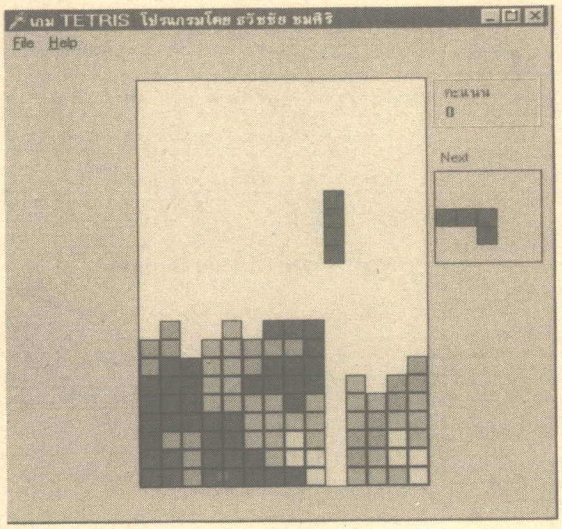
ธวัชชัย ชมศิริ

SOFT MAIL

เขียนเกม TETRIS ด้วย Delphi 3.0



วิธีการเขียนเกม TETRIS โดยใช้โปรแกรม Delphi 3.0

สวัสดีครับท่านผู้อ่าน ยังจำได้ไหมครับเมื่อสมัย 5 ปีที่แล้ว เคยมีเกมหนึ่งที่ฮิตมาก ฮิตจนถึงขนาดที่ว่าเดินไปที่ไหนต้องเจอคนที่กำลังเล่นเกมนี้อยู่ด้วยความเมามัน ไม่ว่าจะเป็นเด็กอนุบาล เด็กประถม มัธยม มหาวิทยาลัย หรือแม้กระทั่งผู้ใหญ่วัยทำงานก็ตาม ใช่แล้วครับ... เกมนี้ก็คือเกม Tetris นั่นเอง

วันนี้ผู้เขียนจะพาท่านผู้อ่านทั้งใหม่และเก่า มารู้จักกับวิธีการสร้างเกมนี้กันนะครับ ซึ่งผู้เขียนได้เคยเขียนเกมนี้ตั้งแต่สมัยที่เรียนอยู่ปี 2 คณะวิทยาศาสตร์ ม.มหาสารคาม ซึ่งเป็นช่วงที่เกมนี้กำลังฮิตสุด ๆ ซึ่งตอนนั้นผู้เขียนใช้ Turbo Pascal 7.0 บนดอส สำหรับวันนี้ผู้เขียนจะพาท่านผู้อ่านมาสร้างเกม Tetris โดยใช้ Delphi 3.0 ผู้อ่านหลายท่านอาจจะสงสัยใช่ไหมครับว่า "Delphi สร้างเกมได้ด้วยหรือ" คำตอบก็คือได้แน่นอนครับ ซึ่งความยากง่ายของเกม และระยะเวลาที่ใช้เขียนนั้นก็แล้วแต่ Logic ของแต่ละคนนะครับ บางท่านอาจจะใช้เวลา 1 ปี บางท่านอาจจะเร็วหน่อยใช้เวลาแค่ไม่กี่เดือน สำหรับผู้เขียนนั้นชอบเกมนี้มากครับจึงใช้เวลาเพียงแค่ 3 ชั่วโมงเท่านั้นเอง

จะเริ่มต้นอย่างไร

ผู้อ่านหลายท่านคงนึกไม่ออกว่าเราจะเริ่มต้นอย่างไรดี จะเขียนคำสั่งอะไรก่อนใช่ไหมครับ

คำตอบคือ เราควรจะพิจารณาลักษณะของเกมที่เราจะเขียนก่อนว่า ควรจะเก็บข้อมูลแบบไหน จะตรวจสอบเหตุการณ์ต่าง ๆ ของเกมได้อย่างไร และวิธีดำเนินการกับตัววัตถุที่อยู่ในเกมทำอย่างไร ใช่แล้วครับเกม Tetris นี้ กรอบสี่เหลี่ยมที่บรรจุวัตถุนั้นลักษณะเป็นแถวและคอลัมน์ ดังนั้นจึงควรจะเป็น Object ที่บรรจุ Array อยู่ข้างใน และตัววัตถุที่มีรูปร่างเป็นกล่องหลายอันต่อกัน และมีการตก การหมุนนั้น ก็ควรจะเป็น Object ที่บรรจุ Array อยู่ข้างในด้วยเช่นกัน สำหรับเหตุการณ์ที่เราต้องตรวจสอบนั้นที่สำคัญ ๆ ก็มี การเคลื่อนที่ไปในทิศทางต่าง ๆ เช่น ซ้าย-ขวา ว่าทำได้หรือไม่ การหมุนทำได้หรือไม่ ตกถึงพื้นหรือยัง สามารถตัดแถวได้หรือยัง และเกมสิ้นสุดหรือยัง เป็นต้น สำหรับวิธีดำเนินการกับวัตถุนั้นที่สำคัญ ๆ ก็เช่น การหมุนและการตัดแถว เป็นต้น

การหมุนของวัตถุ

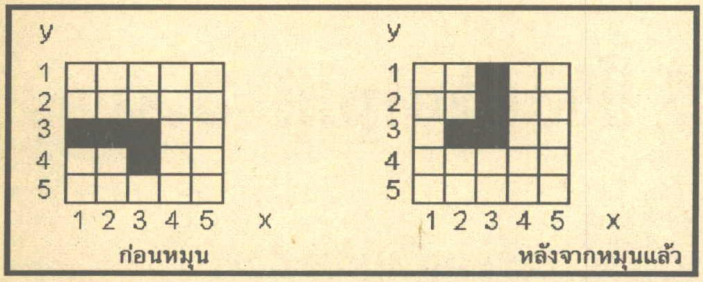
การหมุนของวัตถุนั้น จากรูปที่ 1 ท่านผู้อ่านจะเห็นได้ว่า ค่าที่อยู่ใน Array

```
ตำแหน่ง [x,y] ที่ [1,3] จะเปลี่ยนไปอยู่ในตำแหน่ง [x,y] ที่ [3,1]
ตำแหน่ง [x,y] ที่ [2,3] จะเปลี่ยนไปอยู่ในตำแหน่ง [x,y] ที่ [3,2]
ตำแหน่ง [x,y] ที่ [3,3] จะเปลี่ยนไปอยู่ในตำแหน่ง [x,y] ที่ [3,3]
ตำแหน่ง [x,y] ที่ [3,4] จะเปลี่ยนไปอยู่ในตำแหน่ง [x,y] ที่ [2,3]
```

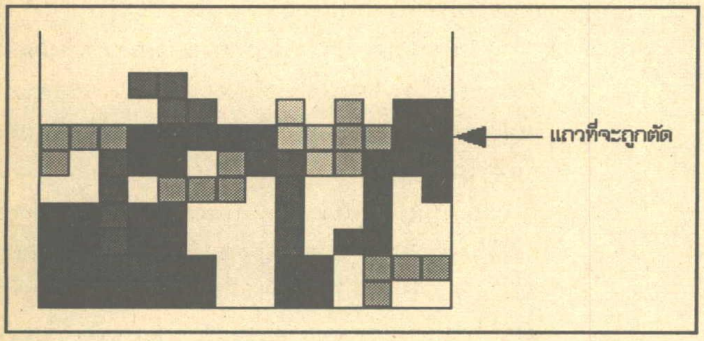
จะเห็นได้ว่า ตำแหน่งที่ [x,y] จะเปลี่ยนไปอยู่ในตำแหน่งที่ [6-y,x] หรือตำแหน่งใหม่ที่ [x,y] จะได้จากตำแหน่งเดิมที่ [y,6-x] นั่นเอง ซึ่งเขียนโค้ดได้ดังนี้

```
for x:=1 to 5 do
for y:=1 to 5 do
WoodTemp.data[x,y]:=Wood.data[y,6-x];
```

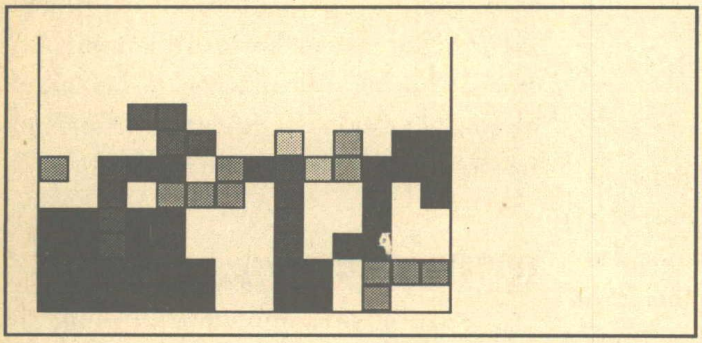
เมื่อให้ WoodTemp เป็น Object ที่เก็บวัตถุที่หมุนแล้ว และ Wood เป็น Object ที่เก็บวัตถุที่ยังไม่หมุน



รูปที่ 1 แสดงรูปวัตถุก่อนทำการหมุนและหลังจากทำการหมุนแล้ว



รูปที่ 2 แสดงแถวที่จะถูกตัด



รูปที่ 3 แสดงรูปหลังจากตัดแถวแล้ว

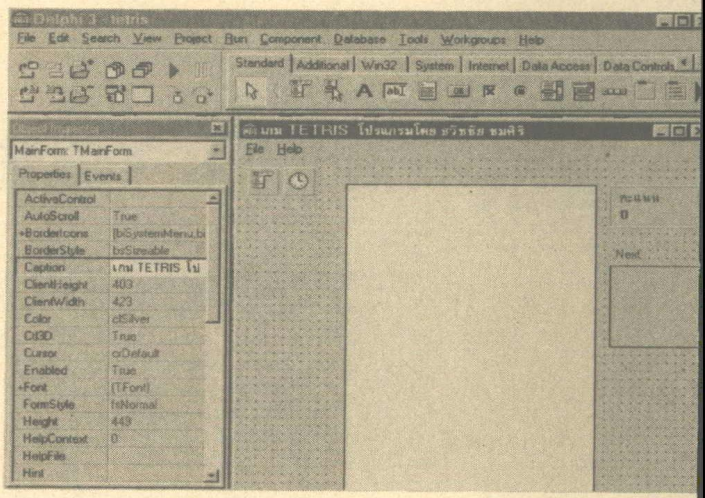
การตัดแถว

การตรวจสอบการตัดแถวนั้นจะตรวจสอบจากการ And กันของตัวแปร Boolean ดังนั้นข้อมูลที่บรรจุใน Array จึงควรจะเป็น Boolean ถ้าแถวใดเต็ม การ And กันทุกตัวในแถวนั้นจะได้ผลเป็น True (ดูรูปที่ 2)

การตัดแถวนั้นก็ทำได้โดย Copy ข้อมูลของแถวข้างบนมาทับแถวที่จะถูกตัด และทำซ้ำไปเรื่อย ๆ จนถึงแถวบนสุด แล้วลบแถวบนสุดให้เป็นแถวว่าง (ดูรูปที่ 3)

มาเริ่มกันเลยดีกว่า

เมื่อท่านผู้อ่านเข้าสู่ Delphi แล้ว ก็ New Application ใหม่ แล้วตั้ง Properties ของ Form นี้



รูปที่ 4 แสดงหน้าจอ Design

```
-Caption='เกม TETRIS'
-Height=449
-Name=MainForm
-Position=poScreenCenter
-Width=431
```

จากนั้นก็ลาก MainMenu1 มาไว้บน Form แล้ว Double Clic
ที่ MainMenu1 แล้วตั้ง Properties ดังนี้

```
-Caption=&File
-Name=File1
```

แล้วสร้าง Menu ย่อย New และ Exit เป็น Menu ย่อยของ Fil
Properties ของ Menu ย่อย New

```
-Caption=&New
-Name= New1
-ShortCut=F2
```

Properties ของ Menu ย่อย Exit

```
-Caption= E&xit
-Name= Exit1
```

สร้างเมนู Help

```
-Caption= &Help
-Name= Help1
```

แล้วสร้าง Menu ย่อย About เป็น Menu ย่อยของ Help แล้วตั้
ค่า Properties ของ About

```
-Caption=&About
-Name= About1
```

ลาก Timer1 มาไว้บน Form แล้วตั้ง Properties ดังนี้

```
-Interval=500
```

ลาก Shape1 มาไว้บน Form แล้วตั้ง Properties ดังนี้

```
-Height=353
-Left=96
-Top=24
-Width=225
```

ลาก Shape2 มาไว้บน Form แล้วตั้ง Properties ดังนี้

```
-Brush.Color= clSilver
-Height=81
-Left=333
-Top=104
-Width=84
```

ลาก Label1 มาไว้บน Form แล้วตั้ง Properties ดังนี้

```
-Caption=Next
-Left=338
-Top=85
```

ลาก Panel1 มาไว้บน Form แล้วตั้ง Properties ดังนี้

```
-Caption=
-Left=332
-Top=24
```

ลาก Label1 มาไว้บน Panel1 แล้วตั้ง Properties ดังนี้

```
-Caption='คะแนน'
-Left=10
-Top=5
```

ลาก Label2 มาไว้บน Panel1 แล้วตั้ง Properties ดังนี้

```
-Caption=0
-Left=10
-Top=23
```

Save File ชื่อ MainFrm และ Save Project ชื่อ Tetris แล้วเขียนซอร์สโค้ดของ Unit MainFrm ดังลิสติงที่ 1

ลิสติงที่ 1

```
unit MainFrm;
interface
uses
 Windows, Messages, SysUtils, Classes,
Graphics, Controls, Forms, Dialogs,
 StdCtrls, Buttons, ExtCtrls, ColorGrd, Menus;
const Wid=16;
 nColumn=14;
 nRow=22;
 BkGroundColor:tColor=clWhite;
 OneLine=100;
 TwoLine=250;
 ThreeLine=500;
 FourLine=1000;
 TimeInterval=500;
type
 TWood = object //ตัววัตถุ
 Xcenter: integer;
 Ycenter: integer;
 Color: TColor;
 Data: array [1..5,1..5] of boolean;

 function MoveDown: boolean;
 function MoveLeft: boolean;
 function MoveRight: boolean;
 function Rotate: boolean;
 procedure SetBeginDataType(nType: integer);
 procedure Show;
 procedure hide;
 function overlab:boolean;
 procedure CopyToBack;
 procedure RandomWood;
 procedure ShowNext;
 private
 BeginDataType: integer;
 public
 end;
 //---------
 TBack = object //กรอบ BackGround
 Data :array[0..nColumn+1,0..nRow+1] of boolean;
 color:array[0..nColumn+1,0..nRow+1] of Tcolor;
 procedure FirstData;
 procedure Show;
 end;
 //---------
 TMainForm = class(TForm)
 Shape1: TShape;
 Timer1: TTimer;
 MainMenu1: TMainMenu;
 File1: TMenuItem;
 New1: TMenuItem;
 Exit1: TMenuItem;
 Help1: TMenuItem;
 About1: TMenuItem;
 Panel1: TPanel;
 Label1: TLabel;
 Label2: TLabel;
 Shape2: TShape;
 Label3: TLabel;
 procedure box(x,y:integer;BrushColors,PenColor:Tcolor);
 procedure FormActivate(Sender: TObject);
 procedure SetFirstValue;
 procedure Timer1Timer(Sender: TObject);
 procedure RanDomWoodType;
 procedure FormKeyDown(Sender: TObject; var Key: Word;
 Shift: TShiftState);
 procedure WoodRotate;
 procedure WoodDown;
 procedure WoodLeft;
 procedure WoodRight;
 procedure New1Click(Sender: TObject);
 procedure Exit1Click(Sender: TObject);
 procedure About1Click(Sender: TObject);
 procedure CheckCut;
 procedure ShowScore;
 procedure CheckGameOver;
 procedure
BoxInShape2(x,y:integer;BrushColors,PenColor:Tcolor);
 private
 { Private declarations }
 public
 { Public declarations }
 end;

var
 MainForm: TMainForm;
 Wood,WoodNext,WoodTemp:TWood; //วัตถุ ตัวปัจจุบัน , ตัวถัดไป , ตัว Temp
 Back:TBack; //กรอบ BackGround
 PlayGame:Boolean=false;
 Falled:Boolean=true;
 Score:LongInt;
 RandomWoodFirst:boolean;
implementation

{$R *.DFM}
procedure TBack.FirstData; //กำหนดค่าเริ่มต้นให้กรอบ BackGround
var x,y:integer;
begin
 for x:=0 to nColumn+1 do
 for y:=0 to nRow+1 do
 begin
 Data[x,y]:=false; //กำหนดให้เป็นที่ว่าง
 Color[x,y]:=BkGroundColor; //กำหนดสีให้เป็นสี BackGround
 end;
```

→

ลิสติงที่ 1 (ต่อ)

```
 for x:=0 to nColumn+1 do
 begin
 Data[x,0]:=true; //ขอบบน
 Data[x,nRow+1]:=true; //ขอบล่าง
 end;
 for y:=0 to nRow+1 do
 begin
 Data[0,y]:=true; //ขอบซ้าย
 Data[nColumn+1,y]:=true; //ขอบขวา
 end;
end;

procedure TBack.Show; //วาดรูปกรอบ BackGround
var x,y:integer;
begin
 for y:=1 to nRow do
 for x:=1 to nColumn do
 if Back.Data[x,y] then
 MainForm.box(x,y,Color[x,y],ClBlack)
 else
 MainForm.box(x,y,Color[x,y],BkGroundColor);
end;

function tocolor(colors:integer):Tcolor; //เปลี่ยนตัวเลขให้เป็นสี
begin
 case colors of
 1:result:=clBlack;
 2:result:=clMaroon;
 3:result:=clGreen;
 4:result:=clOlive;
 5:result:=clNavy;
 6:result:=clPurple;
 7:result:=clTeal;
 8:result:=clGray;
 9:result:=clSilver;
 10:result:=clRed;
 11:result:=clLime;
 12:result:=clBlue;
 13:result:=clFuchsia;
 14:result:=clAqua;
 15:result:=clYellow;
 else result:=clBlack;
 end;
end;

procedure TMainForm.box(x,y:integer;BrushColors,
PenColor:Tcolor);var xx,yy:integer;
begin //วาดสี่เหลี่ยมเล็ก 1 อันใน shape1
 if (x in [1..nColumn]) and (y in [1..nRow]) then
 begin
 dec(x); // ลดค่าลงเพื่อให้เริ่มจาก 0
 dec(y); //
 xx:=(shape1.left+1)+x*wid; // หา Pixel เริ่มต้น
 yy:=(shape1.top+1)+y*wid; //
 canvas.pen.color:=PenColor;
 canvas.Brush.color:=BrushColors;
 canvas.rectangle(xx,yy,xx+wid,yy+wid); //วาดสี่เหลี่ยม
 canvas.floodfill(xx-1,yy-1,clblue,fssurface);
//ระบายสีในสี่เหลี่ยม
 end;
end;

procedure TMainForm.BoxInShape2(x,y:integer;
BrushColors,PenColor:Tcolor);
var xx,yy:integer;
begin //วาดสี่เหลี่ยมเล็ก 1 อันใน shape2
 dec(x);
 dec(y);
 xx:=(shape2.left+1)+x*wid;
 yy:=(shape2.top+1)+y*wid;
 canvas.pen.color:=PenColor;
 canvas.Brush.color:=BrushColors;
 canvas.rectangle(xx,yy,xx+wid,yy+wid);
 canvas.floodfill(xx-1,yy-1,clblue,fssurface)
end;

procedure TWood.Show; //แสดงวัตถุ
var x,y:integer;
begin
 for x:=1 to 5 do
 for y:=1 to 5 do
 if data[x,y] then
 MainForm.box(x+Xcenter-3,y+Ycenter-3,color,clBlack);
end;

procedure TWood.ShowNext; //แสดงวัตถุตัวถัดไปใน Shape2
var x,y:integer;
begin
 MainForm.Shape2.Refresh;
 for x:=1 to 5 do
 for y:=1 to 5 do
 if data[x,y] then
 MainForm.BoxInShape2(x,y,color,clBlack);
end;

procedure TWood.Hide; //ซ่อนวัตถุ
var x,y:integer;
begin
 for x:=1 to 5 do
 for y:=1 to 5 do
 if data[x,y] then
 MainForm.box(x+Xcenter-3,y+Ycenter-
3,BkGroundColor,BkGroundColor);
end;

function TWood.overlab:boolean; //เช็คการเหลื่อมกันของวัตถุ
var x,y:integer;
begin
 result:=false;
 for x:=1 to 5 do
 for y:=1 to 5 do
 if WoodTemp.data[x,y] then //ตรวจสอบจาก Temp กับ BackGround ตลอ
 if Back.Data[x+WoodTemp.Xcenter-3,y+WoodTemp.Ycenter-3]
then result:=true;
end;

procedure TWood.CopyToBack; //Copy ค่าBoolean จากวัตถุไปที่กรอบ
BackGround
var x,y:integer;
begin
 for x:=1 to 5 do
 for y:=1 to 5 do
 if data[x,y] then
 begin
 Back.Data[x+Xcenter-3,y+Ycenter-3]:=True;
 Back.Color[x+Xcenter-3,y+Ycenter-3]:=Color;
 end;
end;

procedure TWood.SetBeginDataType(nType: integer);
//กำหนดรูปร่างวัตถุ
 procedure clearArray5; //Clear ค่าให้ว่างก่อน
 var x,y:integer;
 begin
 for x:=1 to 5 do
 for y:=1 to 5 do
 data[x,y]:=false;
 end;
begin
 case nType of
 1:
 begin
 BeginDataType:=1;
 color:=tocolor(12);
 clearArray5;
 // X X <—รูปร่างของวัตถุหมายเลข 1
 // X X
 data[1,1]:=true;
 data[1,2]:=true;
 data[2,1]:=true;
 data[2,2]:=true;
 end;
 2:
```

→

ลิสติงที่ 1 (ต่อ)

```
begin
BeginDataType:=2;
color:=tocolor(13);
clearArray5;
// X X X X <—รูปร่างของวัตถุหมายเลข 2
data[1,3]:=true; //
data[2,3]:=true; // กำหนดค่าใน Array ให้สัมพันธ์กับรูป
data[3,3]:=true; //
data[4,3]:=true; //
end;
3:
begin
BeginDataType:=3;
color:=tocolor(14);
clearArray5;
// X X X <—รูปร่างของวัตถุหมายเลข 3
// X
data[2,3]:=true;
data[3,3]:=true;
data[4,3]:=true;
data[3,4]:=true;
end;
4:
begin
BeginDataType:=4;
color:=tocolor(11);
clearArray5;
// X X X <—รูปร่างของวัตถุหมายเลข 4
// X
data[3,3]:=true;
data[4,3]:=true;
data[5,3]:=true;
data[3,4]:=true;
end;
5:
begin
BeginDataType:=5;
color:=tocolor(10);
clearArray5;
// X X X <—รูปร่างของวัตถุหมายเลข 5
// X
data[1,3]:=true;
data[2,3]:=true;
data[3,3]:=true;
data[3,4]:=true;
end;
6:
begin
BeginDataType:=6;
color:=tocolor(15);
clearArray5;
// X X <—รูปร่างของวัตถุหมายเลข 6
// X X
data[3,2]:=true;
data[4,2]:=true;
data[2,3]:=true;
data[3,3]:=true;
end;
7:
begin
BeginDataType:=7;
color:=tocolor(7);
clearArray5;
// X X <—รูปร่างของวัตถุหมายเลข 7
// X X
data[2,2]:=true;
data[3,2]:=true;
data[3,3]:=true;
data[4,3]:=true;
end;
end;
end;

function TWood.MoveDown:boolean; //การเคลื่อนตัวลง
begin
 WoodTemp:=Wood; //Copy เก็บไว้ที่ Temp ก่อน
 inc(WoodTemp.Ycenter); //ให้ Temp เคลื่อนที่ลง
 if WoodTemp.overlab then //เมื่อเกิดการเหลื่อมกันระหว่าง Temp กับ
begin //กรอบ BackGround (วัตถุที่ตกลงไปแล้วถือว่า //เป็นส่วนหนึ่งของกรอบ BackGround)
 result:=false; //วัตถุจริงเคลื่อนที่ลงไม่ได้
 CopyToBack; //Copy จากวัตถุจริง ไปที่ กรอบ BackGround
 end
 else //ไม่เกิดการเหลื่อมกัน
 begin
 result:=true; //วัตถุจริงเคลื่อนที่ลงได้
 inc(Ycenter); //ให้วัตถุจริงเคลื่อนที่ลง
 end;
end;

function TWood.MoveLeft:boolean; //ทำในทำนองเดียวกันกับ MoveDown
begin
 WoodTemp:=Wood;
 dec(WoodTemp.Xcenter);
 if WoodTemp.overlab then
 begin
 result:=false;
 end
 else
 begin
 result:=true;
 dec(Xcenter); //เคลื่อนที่ไปทางซ้าย
 end;
end;

function TWood.MoveRight:boolean;//ทำในทำนองเดียวกันกับ MoveDown
begin
 WoodTemp:=Wood;
 inc(WoodTemp.Xcenter);
 if WoodTemp.overlab then
 begin
 result:=false;
 end
 else
 begin
 result:=true;
 inc(Xcenter); //เคลื่อนที่ไปทางขวา
 end;
end;

function TWood.Rotate:boolean; //ทำในทำนองเดียวกันกับ MoveDown
var x,y:integer;
begin
 if Wood.BeginDataType<>1 then
 begin
 WoodTemp:=Wood;
 //—
 for x:=1 to 5 do
 for y:=1 to 5 do
 WoodTemp.data[x,y]:=Wood.data[y,6-x]; //การหมุน
 //—
 if WoodTemp.overlab then
 begin
 result:=false;
 end
 else
 begin
 result:=true;
 Wood:=WoodTemp;
 end;
 end
 else result:=false;
end;

procedure TMainForm.SetFirstValue; //กำหนดค่าเริ่มต้นของ MainForm
begin
 Shape1.Width:=nColumn*wid;
 Shape1.Height:=nRow*wid;
 Shape1.left:=(MainForm.width-Shape1.width) div 2;
 Shape1.top:=24;
 Shape1.left:=Shape1.left-1;
```

ลิสติงที่ 1 (ต่อ)

```
 Shape1.top:=Shape1.top-1;
 Shape1.Width:=Shape1.Width+2;
 Shape1.Height:=Shape1.Height+2;
 Back.FirstData; //กำหนดค่าเริ่มต้นให้กรอบ BackGround
 Score:=0;
end;

procedure TMainForm.FormActivate(Sender: TObject);
begin
 SetFirstValue; //กำหนดค่าเริ่มต้นของ MainForm
 randomize;
end;

procedure TWood.RandomWood;
var r:integer;
begin
 repeat
 r:=random(7)+1; //สุ่มหมายเลขวัตถุจาก 1 ถึง 7
 until r in [1..7];
 SetBeginDataType(r);
 Xcenter:=nColumn div 2;
 case BeginDataType of //วัตถุแต่ละชนิดค่าเริ่มต้นแกน y แตกต่างกัน
 1 :Ycenter:=3;
 2,3,4,5:Ycenter:=1;
 6,7 :Ycenter:=2;
 end;
end;

procedure TMainForm.RanDomWoodType;
begin
 if RandomWoodFirst then //ถ้าเป็นครั้งแรก
 begin
 Wood.RanDomWood; //สุ่มตัวปัจจุบัน
 WoodNext.RanDomWood; //และสุ่มตัวถัดไปด้วย
 RandomWoodFirst:=false;
 end
 else //ไม่ใช่ครั้งแรก
 begin
 Wood:=WoodNext; //นำตัวถัดไปที่สุ่มไว้แล้วมาใช้
 WoodNext.RanDomWood; //สุ่มตัวถัดไป
 end;
 //------
 falled:=false;
 WoodNext.ShowNext;
 Wood.show;
end;

procedure TMainForm.WoodDown;
begin
 wood.hide;
 if wood.MoveDown then //ถ้าเคลื่อนที่ลงได้
 begin
 wood.show;
 end
 else //เคลื่อนที่ลงไม่ได้
 begin
 wood.show;
 RanDomWoodType; //ถือว่าวัตถุตกแล้วให้เริ่มวัตถุตัวถัดไป
 end;
end;

procedure TMainForm.WoodLeft;
begin
 wood.hide;
 wood.MoveLeft;
 wood.show;
end;

procedure TMainForm.WoodRight;
begin
 wood.hide;
 wood.MoveRight;
 wood.show;
end;

procedure TMainForm.WoodRotate;
begin
 wood.hide;
 wood.Rotate;
 wood.show;
end;

procedure TMainForm.CheckCut;
 function IsLine(row:integer):boolean; //เมื่อเรียงกันเป็นแถวในแนวนอน
 var x:integer;
 full:boolean;
 begin
 full:=true;
 for x:=1 to nColumn do
 full:=full and Back.Data[x,row];
 result:=full;
 end;

 procedure cut(row:integer); //ตัดแถว
 var x,y:integer;
 begin
 for y:=row downto 2 do
 for x:=1 to nColumn do
 begin
 Back.Data[x,y]:=Back.Data[x,y-1]; //ย้ายแถวบนลงมาทับ
 Back.Color[x,y]:=Back.Color[x,y-1]; //
 end;
 end;

var nRowCut,y,i:integer;
 RowCut:array[1..4] of integer;
begin
 nRowCut:=0;
 for y:=1 to nRow do //นับจำนวนแถวที่จะตัด
 if IsLine(y) then
 begin
 inc(nRowCut);
 RowCut[nRowCut]:=y;
 end;
 if nRowCut>0 then
 begin
 i:=nRow; //เริ่มตัดจากแถวล่างสุด
 repeat
 if isline(i) then //ถ้าเรียงกันเป็นแถวในแนวนอน,
 begin
 timer1.enabled:=false;
 Cut(i); //ตัด
 timer1.Interval:=timer1.Interval-10;
 timer1.enabled:=true;
 end
 else
 dec(i);
 until i<=2;
 Back.Show;
 end;
 case nRowCut of //ให้คะแนนตามจำนวนแถวที่ตัด
 1:Score:=Score+OneLine;
 2:Score:=Score+TwoLine;
 3:Score:=Score+ThreeLine;
 4:Score:=Score+FourLine;
end;
end;

procedure TMainForm.ShowScore; //แสดงคะแนน
begin
 Label2.Caption:=inttoStr(Score);
 Label2.Refresh;
end;

procedure TMainForm.CheckGameOver; //ตรวจสอบ GemeOver
var x:integer;
 over:boolean;
begin
 over:=false;
```

→

ลิสติงที่ 1 (ต่อ)

```
  for x:=1 to nColumn do
  if Back.Data[x,1] then //ถ้ามีวัตถุค้างในแถวบนสุดคือ GameOver
  over:=true;
  if over then
  begin
  timer1.enabled:=false;
  messagedlg('GAME OVER !'#13+ 'คะแนน='+label2.caption,
  mtinformation,[mbok],0); //------- New Game
  SetFirstValue;
  PlayGame:=true;
  Falled:=true;
  RandomWoodFirst:=true;
  timer1.Interval:=TimeInterval;
  Timer1.enabled:=true;
  end;
 end;

procedure TMainForm.Timer1Timer(Sender: TObject);
begin
  if PlayGame then //ถ้าอยู่ระหว่างการเล่น
  begin
  if Falled then //ถ้าวัตถุตกแล้วก็เริ่มตัวถัดไป
  RanDomWoodType
  else //ถ้ายังไม่ตก
  begin
  WoodDown; //ให้เคลื่อนที่ลง
  CheckCut; //ตรวจสอบการตัด
  ShowScore; //แสดงคะแนน
  CheckGameOver; //ตรวจสอบ GameOver
  end;
  end;
end;

procedure TMainForm.FormKeyDown(Sender: TObject;
var Key: Word;
  Shift: TShiftState);
begin //เมื่อมีการกดคีย์บอร์ด
  if key in [VK_UP, VK_DOWN, VK_LEFT, VK_RIGHT] then
  begin
  case key of
  VK_UP :WoodRotate; //กดลูกศรขึ้น
  VK_DOWN :WoodDown; //กดลูกศรลง
  VK_LEFT :WoodLeft; //กดลูกศรซ้าย
  VK_RIGHT:WoodRight; //กดลูกศรขวา
  end;
  key:=0;
  end;
end;

procedure TMainForm.New1Click(Sender: TObject);
begin
  //--New Game
  Timer1.enabled:=false;
  SetFirstValue; //กำหนดค่าเริ่มต้น
  PlayGame:=true;
  Falled:=true;
  RandomWoodFirst:=true;
  timer1.Interval:=TimeInterval;
  Timer1.enabled:=true;
end;

procedure TMainForm.Exit1Click(Sender: TObject);
begin
  close;
end;

procedure TMainForm.About1Click(Sender: TObject);
begin
  messagedlg('เกม TETRIS'#13'โปรแกรมโดย ธวัชชัย ชมศิริ',
mtinformation,[mbok],0);
end;
end.
```

คอมพิวเตอร์คาราโอเกะ

ปัจจุบันนี้เรามักเห็นมีการเปิดคาราโอเกะด้วยคอมพิวเตอร์ มีเสียงเพลงและตัวหนังสือวิ่งไปตามเสียงเพลง เพื่อให้ได้ร้องเพลงกันตามตัวหนังสือเหมือนในวีดีโอคาราโอเกะ เพียงแต่ไม่มีรูปภาพ หลายคนเห็นแล้วก็อยากจะนำมาเล่นบ้าง สอบถามแล้วก็ได้ความว่าเป็นโปรแกรมอยู่บนแผ่น CD ทำให้คิดว่าเครื่องของเราก็มีไดรฟ์ CD คงจะนำมาเล่นได้ แต่เอาเข้าจริง ๆ กลับเล่นไม่ได้ ทั้งที่ลองเล่น midi file จากแผ่นก็เล่นได้ แต่เล่นคาราโอเกะไม่ได้ ที่เล่นไม่ได้ก็เพราะว่าฮาร์ดแวร์ไม่รองรับนั่นเอง ซึ่งได้แก่ส่วนของซาวน์การ์ด (Sound Card) การ์ดเสียงจะเล่นคาราโอเกะได้จะต้องมี Wavetable ในตัว บางคนว่าเครื่องของผมก็เล่น wave file ได้คงจะมี Wavetable ในตัวเหมือนกัน ไม่ใช่ครับ ถ้ามี Wavetable ก็สามารถเล่นคาราโอเกะได้เลย การ์ดรุ่นใหม่ที่ต่อท้ายด้วย AWE มักจะเล่นคาราโอเกะได้เลย แต่ถ้าเป็นการ์ดรุ่นเก่า ก็ต้องเปิดเครื่องดูเอาล่ะครับว่าการ์ดตัวนั้นมีช่องสำหรับเสียบอุปกรณ์เสริมได้หรือเปล่า

ช่องสำหรับเสียบอุปกรณ์เสริมจะเป็นขาเล็ก ๆ สองแถว จำนวน 26 ขา ถ้าการ์ดมีขาลักษณะแบบนี้แสดงว่าสามารถเพิ่มอุปกรณ์เพื่อเล่นคาราโอเกะได้ แต่ถ้าไม่มีก็หมดสิทธิ์ครับ สำหรับการ์ดที่มีช่องเสียบเราจำเป็นต้องซื้อการ์ดเล็ก ๆ มาเสียบ การ์ดตัวนี้มีชื่อว่า Wavetable Module เมื่อนำมาเสียบบนซาวน์การ์ดแล้วก็จะเป็นการเพิ่มประสิทธิภาพซาวน์การ์ดให้สามารถเล่นคาราโอเกะได้ ขอให้สนุกกับการร้องเพลงนะครับ...

โดย คุณวรกิจ วิริยะเกษามงคล
วิทยาลัยเทคนิคชลบุรี
ต.หนองชาก อ.บ้านบึง
จ.ชลบุรี 20170
icq 3909077

(เจ้าของเทคนิคนี้ จะได้รับโปรแกรม Thaisoft Dictionary เวอร์ชัน 2.0 จำนวน 1 ชุด)